REMARKS

In the Official Action mailed on **23 May 2008**, Examiner reviewed claims 1-2, 4-11, and 13-20. Examiner rejected claims 1-2, 4-11 and 13-20 under 35 U.S.C. § 112. Examiner rejected claims 1-2, 4-7, 9-11, 13-16, and 18-20 under 35 U.S.C. § 103(a) based on Rajwar et al. (U.S. Patent No. 7,120,726, hereinafter "Rajwar"), in view of Goodman et al. (*"Speculative Lock Elison: Enabling Highly Concurrent Multithreaded Execution"*, IEEE 2001, hereinafter "Goodman"), and in further view of Wang et al. (U.S. Patent No. 6,006,299, hereinafter "Wang"). Examiner rejected claims 8 and 17 under 35 U.S.C. § 103(a) as being unpatentable over Rajwar, in view of Goodman, in further view of Wang, and in further view Hecht et al. (U.S. Pub. No. 2003/0064808, hereinafter "Hecht").

Rejections under 35 U.S.C. § 112

Examiner rejected claims 1-2, 4-11, 13-20 under 35 U.S.C. § 112 for failing to particularly point out and distinctly claim the inventive subject matter. Specifically, Examiner indicated that the claim language did not sufficiently explain the differences between store-marking and load-marking cache lines.

Applicant has amended claims 1, 10, and 19 to clarify that embodiments of the present invention load-mark a cache line when data is loaded from the cache line, and store-mark a cache line when data is stored to the cache line. These amendments are supported in par. [0053] and par. [0056] of the instant application. No new matter was added. These amendments render moot the claim rejections under 35 U.S.C. § 112, therefore Applicant respectfully requests that the claim rejections be withdrawn.

Rejections under 35 U.S.C. 103(a)

Examiner rejected claims 1, 2, 4-7, 9-11, 13-16, and 18-20 under 35 U.S.C. § 103(a) as being obvious under Rajwar, in view of Goodman, and in further view of Wang. The rejections under 35 U.S.C. § 103(a) are addressed in the following sections.

Load-Marking and Store-Marking

Examiner averred that Goodman discloses atomically committing changes by marking cache lines. Specifically, Examiner averred that the access bit of Goodman corresponds to the load-marks and store-marks on the cache line of embodiments of the present invention (see Office Action, page 4). Applicant respectfully disagrees. Nowhere do Rajwar, Goodman, or Wang, either separately or combined, disclose that executing the block of instructions transactionally involves placing load-marks on cache lines from which data is loaded and placing store-marks on cache lines to which data is stored.

Goodman describes "Speculative Lock Elision" (SLE), a technique that involves dynamically removing locks from critical sections of multithreaded code and then speculatively executing the code (see Goodman, page 297, right column). Goodman's system *predicts* that memory operations in critical code sections will execute atomically, removes synchronization code from the critical code sections, and then allows multiple threads to speculatively (and potentially simultaneously) execute the critical code sections (see Goodman, page 297, left column). Goodman discloses an access bit for every cache block to detect data conflicts (see Goodman, page 297, left column). In the system of Goodman, the access bit for a cache block is set for every cache line that is accessed during speculative execution (see Goodman, page 299, right column, to page 300, left column, paragraph 2, beginning with "*If the register checkpoint approach is used*"). In

other words, the system of Goodman uses **a single bit for tracking *both* load and store operations during speculative execution**. Nowhere does Goodman disclose that executing the block of instructions transactionally involves placing load-marks on cache lines from which data is loaded and placing store-marks on cache lines to which data is stored.

In contrast, embodiments of the present invention use two separate markings to distinguish between cache lines that are loaded from and stored to. The load marking bit of a cache line indicates that data was loaded from the cache line, and the store marking bit indicates that data was stored to the cache line (see instant application, par. [0053], and [0056]). Maintaining two separate marking bits allows embodiments of the present invention to distinguish between load-marked and store-marked cache lines, and then take different actions during a commit operation at the end of transactional execution, depending on which marking is set (see instant application, pars. [0077]-[0078]).

Note that **load-marks** also enable other threads or processors to read from the cache line, but not to write to the cache line. On the other hand, **store-marks** prevent other threads or processors from either reading from or writing to a store-marked cache line. Goodman nowhere describes using marks to enable such cache line access patterns.

Applicant has amended independent claims 1, 10, and 19 to clarify that embodiments of the present invention load-mark a cache line if data is loaded from the cache line during transactional execution and store-mark a cache line if data is stored to the cache line during transactional execution. Nowhere do Rajwar, Goodman, or Wang, either separately or combined, disclose that executing the block of instructions transactionally involves placing load-marks on cache lines from which data is loaded and placing store-marks on cache lines to which data is stored.

For this reason, Applicant respectfully submits that the rejection under 35 U.S.C. § 103(a) is incorrect and requests the withdrawal of the rejection.

Locked Cache Lines

Examiner averred that Goodman discloses treating marked cache lines as locked; causing other processes to wait to access the store-marked cache lines (see Office Action, page 4). Applicant respectfully disagrees. Nowhere do Rajwar, Goodman, or Wang, either separately or combined, disclose treating store-marked cache lines as locked, thereby causing other processes to wait to access the store-marked cache-lines.

The system of Goodman marks cache lines during SLE to record the access (read or write) of the cache line. However, Goodman **does nothing to prevent another process from subsequently accessing the marked cache line** (i.e., Goodman allows more than one process to access marked cache lines during SLE) (see Goodman, page 297, left column). Goodman simply discloses using the access marks on the cache lines to *detect data conflicts during speculative execution*. If a conflict is detected, the system of Goodman cancels SLE and reverts to the traditional concurrent execution using locks (see Goodman, page 298, left column). In other words, the system of Goodman relies on threads executing concurrently and possibly accessing marked cache lines at the same time, ***without locking or waiting*** in order to detect SLE mispredictions. Nothing in Goodman discloses treating store-marked cache lines as locked, thereby **causing other processes to wait to access the store-marked cache lines**.

In contrast, embodiments of the present invention use a load-marking bit and a store-marking bit to distinguish between store and load memory accesses during transaction execution, which allows embodiments of the present invention to perform different operations for load-marked and stored-marked cache lines.

For example, in some embodiments of the present invention, load-marked lines are cleared, but store-marked cache lines are locked during the commit operation, causing other processors to wait to access the store-marked cache lines (see instant application, par. [0077]-[0078]).

Note that **load-marks** also enable other threads or processors to read from the cache line, but not to write to the cache line. On the other hand, **store-marks** prevent other threads or processors from either reading from or writing to a store-marked cache line. Goodman nowhere describes using marks to enable such cache line access patterns. For this reason, Applicant respectfully submits that the rejection under 35 U.S.C. § 103(a) is incorrect and requests the withdrawal of the rejection.

Hence, Applicant respectfully submits that independent claims 1, 10, and 19 as presently amended are in condition for allowance. Applicant also submits that the dependent claims that depend upon these independent claims are in condition for allowance and for reasons of the unique combinations recited in such claims.

CONCLUSION

It is submitted that the application is presently in form for allowance. Such action is respectfully requested.

Respectfully submitted,

By /Anthony Jones/
Anthony Jones
Registration No. 59,521

Date: 25 August 2008

Anthony Jones
Park, Vaughan & Fleming LLP
2820 Fifth Street
Davis, CA 95618-7759
Tel: (530) 759-1666
Fax: (530) 759-1665
Email: tony@parklegal.com